K 7

```
RRRRRRRRRRRR     UUU         UUU  NNN         NNN     OOOOOOOOO     FFFFFFFFFFFFFFF  FFFFFFFFFFFFFFF
RRRRRRRRRRRR     UUU         UUU  NNN         NNN     OOOOOOOOO     FFFFFFFFFFFFFFF  FFFFFFFFFFFFFFF
RRRRRRRRRRRR     UUU         UUU  NNN         NNN     OOOOOOOOO     FFFFFFFFFFFFFFF  FFFFFFFFFFFFFFF
RRR         RRR  UUU         UUU  NNN         NNN  OOO         OOO  FFF              FFF
RRR         RRR  UUU         UUU  NNN         NNN  OOO         OOO  FFF              FFF
RRR         RRR  UUU         UUU  NNN         NNN  OOO         OOO  FFF              FFF
RRR         RRR  UUU         UUU  NNNNNN      NNN  OOO         OOO  FFF              FFF
RRR         RRR  UUU         UUU  NNNNNN      NNN  OOO         OOO  FFF              FFF
RRR         RRR  UUU         UUU  NNNNNN      NNN  OOO         OOO  FFF              FFF
RRRRRRRRRRRR     UUU         UUU  NNN   NNN   NNN  OOO         OOO  FFFFFFFFFFFF     FFFFFFFFFFFF
RRRRRRRRRRRR     UUU         UUU  NNN   NNN   NNN  OOO         OOO  FFFFFFFFFFFF     FFFFFFFFFFFF
RRRRRRRRRRRR     UUU         UUU  NNN   NNN   NNN  OOO         OOO  FFFFFFFFFFFF     FFFFFFFFFFFF
RRR   RRR        UUU         UUU  NNN      NNNNNN  OOO         OOO  FFF              FFF
RRR   RRR        UUU         UUU  NNN      NNNNNN  OOO         OOO  FFF              FFF
RRR   RRR        UUU         UUU  NNN      NNNNNN  OOO         OOO  FFF              FFF
RRR      RRR     UUU         UUU  NNN         NNN  OOO         OOO  FFF              FFF
RRR      RRR     UUU         UUU  NNN         NNN  OOO         OOO  FFF              FFF
RRR      RRR     UUU         UUU  NNN         NNN  OOO         OOO  FFF              FFF
RRR         RRR  UUUUUUUUUUUUUUU  NNN         NNN     OOOOOOOOO     FFF              FFF
RRR         RRR  UUUUUUUUUUUUUUU  NNN         NNN     OOOOOOOOO     FFF              FFF
RRR         RRR  UUUUUUUUUUUUUUU  NNN         NNN     OOOOOOOOO     FFF              FFF
```

```
_$2

Sym
---
NDX
NDX
NUM
NUM
OPE
OUT
PAC
PAC
PAC
PAC
PAC
PAC
PAC
PAD
PAG
PAG
PAG
PAG
PAG
PAG
PAG
PER
PUT
RCO
RIN

RLI
RNO
RNO
RTY
SAV
STR
STR
STR
STR
STR
STR

STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
STR
```

```
**FILE**ID**DOFLG                                                  D  5





DDDDDDDD      000000    FFFFFFFFFF  LL            GGGGGGGG
DDDDDDDD      000000    FFFFFFFFFF  LL            GGGGGGGG
DD      DD  00      00  FF          LL          GG
DD      DD  00      00  FF          LL          GG
DD      DD  00      00  FF          LL          GG
DD      DD  00      00  FF          LL          GG
DD      DD  00      00  FFFFFFFF    LL          GG
DD      DD  00      00  FFFFFFFF    LL          GG
DD      DD  00      00  FF          LL          GG  GGGGGG
DD      DD  00      00  FF          LL          GG  GGGGGG
DD      DD  00      00  FF          LL          GG      GG      ....
DD      DD  00      00  FF          LL          GG      GG      ....
DDDDDDDD      000000    FF          LLLLLLLLLL    GGGGGG        ....
DDDDDDDD      000000    FF          LLLLLLLLLL    GGGGGG        ....

LL            IIIIII      SSSSSSSS
LL            IIIIII      SSSSSSSS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II        SSSSSS
LL              II        SSSSSS
LL              II              SS
LL              II              SS
LL              II              SS
LL              II              SS
LLLLLLLLLL    IIIIII    SSSSSSSS
LLLLLLLLLL    IIIIII    SSSSSSSS
```

```
    0001  0 %TITLE 'Process flags'
    0002  0 MODULE DOFLG ( IDENT = 'V04-000'
P   0003  0                 %BLISS32 [ , ADDRESSING_MODE ( EXTERNAL =      LONG_RELATIVE,
    0004  0                                                NONEXTERNAL = LONG_RELATIVE)]
    0005  0                 ) =
    0006  1 BEGIN
    0007  1 !
    0008  1 !****************************************************************************
    0009  1 !*                                                                          *
    0010  1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
    0011  1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
    0012  1 !*  ALL RIGHTS RESERVED.                                                    *
    0013  1 !*                                                                          *
    0014  1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
    0015  1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
    0016  1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
    0017  1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
    0018  1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
    0019  1 !*  TRANSFERRED.                                                            *
    0020  1 !*                                                                          *
    0021  1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
    0022  1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
    0023  1 !*  CORPORATION.                                                            *
    0024  1 !*                                                                          *
    0025  1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
    0026  1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
    0027  1 !*                                                                          *
    0028  1 !*                                                                          *
    0029  1 !****************************************************************************
    0030  1 !
    0031  1 !++
    0032  1 ! FACILITY:      DSR (Digital Standard RUNOFF) / DSRPLUS
    0033  1 !
    0034  1 ! ABSTRACT: Flag processing
    0035  1 !
    0036  1 !
    0037  1 ! ENVIRONMENT: Transportable
    0038  1 !
    0039  1 ! AUTHOR: R.W.Friday
    0040  1 !
    0041  1 ! CREATION DATE: April 1978
    0042  1 !
```

```
1 %SBTTL 'Revision History'
1 !
1 ! MODIFIED BY:
1 !
1 !       020     REM00020        Ray Marshall    17-November-1983
1 !               Added supporting logic for the case rules for the DEC multi-
1 !                 national character set.
1 !
1 !       019     KFA00019        Ken Alden       06-Oct-1983
1 !               For DSRPLUS: Removed call to endwrd during pass-through
1 !
1 !       018     KFA00018        Ken Alden       13-Sep-1983
1 !               For DSRPLUS: Added sca_wrd_pass for ENDWRD cooperation.
1 !
1 !       017     KFA00017        Ken Alden       11-Jun-1983
1 !               Fixed the bolded, underline TAB bug.
1 !
1 !       016     KFA00016        Ken Alden       21-Mar-1983
1 !               Removed error message RNFIPF.
1 !
1 !       015     KFA00015        Ken Alden       21-Mar-1983
1 !               Eliminate the Passthrough flag, functionality now works
1 !               with accept flag
1 !
1 !       014     KAD00014        Keith Dawson    20-Mar-1983
1 !               Fixed bug: passthrough was visible to DSR.
1 !
1 !       013     KFA00013        Ken Alden       15-Mar-1983
1 !               For DSRPLUS: added recognition of passthrough flag
1 !
1 !       012     KAD00012        Keith Dawson    07-Mar-1983
1 !               Global edit of all modules. Updated module names, idents,
1 !               copyright dates. Changed require files to BLISS library.
1 !
1 !--
```

```
;     80            0078    1 %SBTTL 'Module Level Declarations'
;     81            0079    1
;     82            0080    1 !
;     83            0081    1 ! TABLE OF CONTENTS:
;     84            0082    1 !
;     85            0083    1
;     86            0084    1 FORWARD ROUTINE
;     87            0085    1     SUBXR : NOVALUE,                                              !Used by .SUBINDEX and .ENTRY commands
;     88            0086    1     XR : NOVALUE;                                                 !Used for processing Index flag
;     89            0087    1
;     90            0088    1 !
;     91            0089    1 ! INCLUDE FILES:
;     92            0090    1 !
;     93            0091    1
;     94            0092    1 LIBRARY 'NXPORT:XPORT';                           ! XPORT Library
;     95            0093    1 REQUIRE 'REQ:RNODEF';                             ! RUNOFF variant definitions
;     96            0224    1
;     97          U 0225    1 %IF DSRPLUS %THEN
;     98          U 0226    1 LIBRARY 'REQ:DPLLIB';                             ! DSRPLUS BLISS Library
;     99            0227    1 %ELSE
;    100            0228    1 LIBRARY 'REQ:DSRLIB';                             ! DSR BLISS Library
;    101            0229    1 %FI
;    102            0230    1
;    103            0231    1 !
;    104            0232    1 ! MACROS:
;    105            0233    1 !
;    106            0234    1 !
;    107            0235    1 ! EQUATED SYMBOLS:
;    108            0236    1 !
;    109            0237    1 !
;    110            0238    1 ! OWN STORAGE:
;    111            0239    1 !
;    112            0240    1 !
;    113            0241    1 ! EXTERNAL REFERENCES:
;    114            0242    1 !
;    115            0243    1
;    116            0244    1 EXTERNAL LITERAL
;    117            0245    1     RINTES : UNSIGNED (8);
;    118            0246    1
;    119            0247    1 EXTERNAL
;    120            0248    1     FLGT : FLAG_TABLE [FLAG_COUNT],
;    121            0249    1     IRA : FIXED_STRING,
;    122            0250    1     KHAR,
;    123            0251    1     MRA : REF FIXED_STRING,
;    124            0252    1     SCA : SCA_DEFINITION,
;    125            0253    1     TSF : TSF_DEFINITION,
;    126            0254    1     XMRA : FIXED_STRING,
;    127            0255    1     XTSF : VECTOR;
;    128            0256    1 !
;    129            0257    1 EXTERNAL LITERAL                                                  ! Error messages
;    130            0258    1     RNFFEL,
;    131            0259    1     RNFFNA,
;    132            0260    1     RNFLOC,
;    133            0261    1     RNFLTC,
;    134            0262    1     RNFSTR;
;    135            0263    1
;    136            0264    1 EXTERNAL ROUTINE
;   137          0265   1       endchr,
;   138          0266   1       endwrd,
;   139          0267   1       erm,
;   140          0268   1       erms,
;   141          0269   1       fndflg,
;   142          0270   1       outlin,
;   143          0271   1       outxph,
;   144        U 0272   1 %IF dsrplus %THEN
;   145        U 0273   1       newsub,
;   146          0274   1 %ELSE
;   147          0275   1       subst,
;   148          0276   1 %FI
;   149          0277   1       rskips,
;   150          0278   1       setcas;
;  152          0279  1 GLOBAL ROUTINE DOFLG : NOVALUE =
;  153          0280  1
;  154          0281  1 !++
;  155          0282  1 ! FUNCTIONAL DESCRIPTION:
;  156          0283  1 !
;  157          0284  1 !
;  158          0285  1 ! FORMAL PARAMETERS: None
;  159          0286  1 !
;  160          0287  1 ! IMPLICIT INPUTS:
;  161          0288  1 !
;  162          0289  1 !       KHAR contains the character (flag) to be processed.
;  163          0290  1 !
;  164          0291  1 ! IMPLICIT OUTPUTS: None
;  165          0292  1 !
;  166          0293  1 ! ROUTINE VALUE:
;  167          0294  1 ! COMPLETION CODES: None
;  168          0295  1 !
;  169          0296  1 ! SIDE EFFECTS: None
;  170          0297  1 !
;  171          0298  1 !--
;  172          0299  1
;  173          0300  2     BEGIN
;  174          0301  2
;  175          0302  2     OWN
;  176          0303  2         SCA_DO_HOLD,
;  177          0304  2         SCA_WRD_CHOLD;
;  178          0305  2
;  179          0306  2     LOCAL
;  180          0307  2         WHICH_FLAG;
;  181          0308  2
;  182          0309  2     WHICH_FLAG = FNDFLG (.KHAR);                    !Identify the flag to be processed.
;  183          0310  2
;  184          0311  2     CASE .WHICH_FLAG FROM 0 TO FLAG_COUNT - 1 OF
;  185          0312  2         SET
;  186          0313  2
;  187          0314  2         [SUB_FLAG] :
;  188          0315  3             BEGIN
;  189          0316  3             !Substitute flag
;  190        U 0317  3 %IF dsrplus %THEN
;  191        U 0318  3             newsub ();
;  192          0319  3 %ELSE
;  193          0320  3             SUBST ();
;  194          0321  3 %FI
;  195          0322  3             RETURN;
;  196          0323  2             END;
;  197          0324  2
;  198          0325  2         [QUO_FLAG] :
;  199          0326  3             BEGIN
;  200          0327  3             !Quote flag
;  201          0328  3             KCNS ();                                !Get next character.
;  202          0329  3
;  203          0330  3             IF .KHAR EQL RINTES
;  204          0331  3             THEN
;  205          0332  4                 BEGIN
;  206          0333  4                 ERM (RNFFEL, CH$PTR (UPLIT ('ACCEPT')), 6);
;  207          0334  4                 ERMS  (RNFSTR, .FS_START (IRA), CH$DIFF (.FS_NEXT (IRA), .FS_START (IRA)));
;  208          0335  4                 RETURN;                         ! Ignore a Quote flag at the end of the line.
;  209        0336  4                 END
;  210        0337  3             ELSE                                    !
;  211        0338  4                 BEGIN
;  212        0339  4                 SCA_FRC_CHR = TRUE;                 !Don't translate this character.
;  213        0340  4                 ENDCHR (.KHAR);                     !Output character as is.
;  214        0341  4                 KCNS ();                            !Get next character.
;  215        0342  4                 RETURN;
;  216        0343  3                 END;
;  217        0344  3
;  218        0345  2             END;
;  219        0346  2
;  220        0347  2         [UPP_FLAG] :
;  221        0348  3             BEGIN
;  222        0349  3             !Uppercase flag
;  223        0350  3             KCNS ();                                !Get next character.
;  224        0351  3
;  225        0352  3             IF .KHAR EQL .FLGT [UPP_FLAG, FLAG_CHARACTER]
;  226        0353  3             THEN
;  227        0354  4                 BEGIN                               !Doubled Uppercase flags
;  228        0355  4                 SETCAS (LEAVE_CASE);
;  229        0356  4                 SCA_FRC_CASE = TRUE;
;  230        0357  4                 KCNS ();
;  231        0358  4                 RETURN;
;  232        0359  3                 END;
;  233        0360  3
;  234        0361  4             IF (.KHAR EQL .FLGT [CAP_FLAG, FLAG_CHARACTER])
;  235        0362  3                 AND .FLGT [CAP_FLAG, FLAG_ENABLED]
;  236        0363  3             THEN
;  237        0364  3             !Uppercase flag followed by Capitalize flag (i.e., ^<)
;  238        0365  4                 BEGIN
;  239        0366  4                 SETCAS (FORCE_UPPER);
;  240        0367  4                 SCA_FRC_CASE = TRUE;
;  241        0368  4                 KCNS ();
;  242        0369  4                 RETURN
;  243        0370  3                 END;
;  244        0371  3
;  245        0372  4             IF (.KHAR EQL .FLGT [UND_FLAG, FLAG_CHARACTER])
;  246        0373  3                 AND .FLGT [UND_FLAG, FLAG_ENABLED]
;  247        0374  3             THEN
;  248        0375  3             !Uppercase flag followed by Underline flag (i.e., ^&)
;  249        0376  4                 BEGIN                               !Turn on global underlining
;  250        0377  4                 SCA_UND = TRUE;
;  251        0378  4                 SCA_WRD_C_UND = .SCA_DO_UND;
;  252        0379  4                 KCNS ();
;  253        0380  4                 RETURN;
;  254        0381  3                 END;
;  255        0382  3
;  256        0383  4             IF (.KHAR EQL .FLGT [BLD_FLAG, FLAG_CHARACTER])
;  257        0384  3                 AND .FLGT [BLD_FLAG, FLAG_ENABLED]
;  258        0385  3             THEN
;  259        0386  3             !Uppercase flag followed by Bold flag (i.e., ^*)
;  260        0387  4                 BEGIN                               !Turn on global bolding
;  261        0388  4                 SCA_BLD = TRUE;
;  262        0389  4                 SCA_WRD_C_BLD = .SCA_DO_BLD;
;  263        0390  4                 KCNS ();
;  264        0391  4                 RETURN;
;  265        0392  3                 END;
;   266            0393  3
;   267          U 0394  3  %IF DSRPLUS %THEN
;   268          U 0395  3              IF (.KHAR EQL .FLGT [QUO_FLAG, FLAG_CHARACTER])
;   269          U 0396  3                  AND .FLGT [QUO_FLAG, FLAG_ENABLED]
;   270          U 0397  3              THEN
;   271          U 0398  3              !Uppercase flag followed by Accept flag (i.e., ^ )
;   272          U 0399  3                  BEGIN                               !Turn on global accept
;   273          U 0400  3                  sca_pass = true;                    !Now passing through (ENDCHR)
;   274          U 0401  3                  sca_wrd_pass = true;                !This 'word' is a passthr. (ENDWRD)
;   275          U 0402  3                  sca_do_hold = .sca_do_nbits;        !Save status of bold/under
;   276          U 0403  3                  sca_wrd_chold = .sca_wrd_cnbits;    !
;   277          U 0404  3                  sca_do_nbits = false;               !and turn it temp. off.
;   278          U 0405  3                  sca_wrd_cnbits = false;             !
;   279          U 0406  3                  KCNS ();
;   280          U 0407  3                  RETURN;
;   281          U 0408  3                  END;
;   282          U 0409  3
;   283            0410  3  %FI
;   284            0411  3              !This gets done for an Uppercase flag standing alone.
;   285            0412  3              SCA_FRC_CHR = TRUE;
;   286            0413  3              SCA_WRD_FC_LT = %C'A' - %C'a';
;   287            0414  3              SCA_WRD_OC_LT = %C'A' - %C'a';
;   288            0415  3              SCA_WRD_FC_UT = 0;
;   289            0416  3              SCA_WRD_OC_UT = 0;
;   290            0417  3
;   291            0418  3              SCA_MNWRD_FC_LT = %DECIMAL'192' - %DECIMAL'224';
;   292            0419  3              SCA_MNWRD_OC_LT = %DECIMAL'192' - %DECIMAL'224';
;   293            0420  3              SCA_MNWRD_FC_UT = 0;
;   294            0421  3              SCA_MNWRD_OC_UT = 0;
;   295            0422  3              RETURN
;   296            0423  2              END;
;   297            0424  2
;   298            0425  2          [LOW_FLAG] :
;   299            0426  3              BEGIN
;   300            0427  3              !Lowercase flag
;   301            0428  3              KCNS ();                                !Get next character.
;   302            0429  3
;   303            0430  3              IF .KHAR EQL .FLGT [LOW_FLAG, FLAG_CHARACTER]
;   304            0431  3              THEN
;   305            0432  4                  BEGIN                               ! Doubled Lowercase flags
;   306            0433  4                  SETCAS (FORCE_LOWER);
;   307            0434  4                  SCA_FRC_CASE = TRUE;
;   308            0435  4                  KCNS ();
;   309            0436  4                  RETURN;
;   310            0437  4                  END;
;   311            0438  3
;   312            0439  4              IF (.KHAR EQL .FLGT [UND_FLAG, FLAG_CHARACTER])
;   313            0440  3                  AND .FLGT [UND_FLAG, FLAG_ENABLED]
;   314            0441  3              THEN
;   315            0442  3              !Lowercase flag followed by Underline flag (i.e., \&)
;   316            0443  4                  BEGIN                               !Turn off global underlining
;   317            0444  4                  SCA_UND = FALSE;
;   318            0445  4                  SCA_WRD_C_UND = FALSE;
;   319            0446  4                  KCNS ();
;   320            0447  4                  RETURN;
;   321            0448  3                  END;
;   322            0449  3
;  323          0450  4             IF (.KHAR EQL .FLGT [BLD_FLAG, FLAG_CHARACTER])
;  324          0451  3                 AND .FLGT [BLD_FLAG, FLAG_ENABLED]
;  325          0452  3             THEN
;  326          0453  3             !Lowercase flag followed by Bold flag (i.e., \*)
;  327          0454  4                 BEGIN                          !Turn off global bolding
;  328          0455  4                 SCA_BLD = FALSE;
;  329          0456  4                 SCA_WRD_C_BLD = FALSE;
;  330          0457  4                 KCNS ();
;  331          0458  4                 RETURN;
;  332          0459  3                 END;
;  333          0460  3
;  334        U 0461  3 %IF DSRPLUS %THEN
;  335        U 0462  3             IF (.KHAR EQL .FLGT [QUO_FLAG, FLAG_CHARACTER])
;  336        U 0463  3                 AND .FLGT [QUO_FLAG, FLAG_ENABLED]
;  337        U 0464  3             THEN
;  338        U 0465  3             !Lowercase flag followed by Accept flag (i.e., \_)
;  339        U 0466  3                 BEGIN                          !Turn off global Accept
;  340        U 0467  3                 IF .SCA_PASS EQL TRUE
;  341        U 0468  3                 THEN
;  342        U 0469  3                     BEGIN
;  343        U 0470  3                     SCA_DO_NBITS = .SCA_DO_HOLD;         !Restore status of bold/under
;  344        U 0471  3                     SCA_WRD_CNBITS = .SCA_WRD_CHOLD;     !
;  345        U 0472  3                     END;
;  346        U 0473  3                 SCA_PASS = FALSE;
;  347        U 0474  3                 KCNS ();
;  348        U 0475  3                 RETURN;
;  349        U 0476  3                 END;
;  350          0477  3 %FI
;  351          0478  3             !Lowercase flag standing alone
;  352          0479  3             SCA_FRC_CHR = TRUE;
;  353          0480  3             SCA_WRD_FC_UT = %C'a' - %C'A';
;  354          0481  3             SCA_WRD_OC_UT = %C'a' - %C'A';
;  355          0482  3             SCA_WRD_FC_LT = 0;
;  356          0483  3             SCA_WRD_OC_LT = 0;
;  357          0484  3
;  358          0485  3             SCA_MNWRD_FC_UT = %DECIMAL'224' - %DECIMAL'192';
;  359          0486  3             SCA_MNWRD_OC_UT = %DECIMAL'224' - %DECIMAL'192';
;  360          0487  3             SCA_MNWRD_FC_LT = 0;
;  361          0488  3             SCA_MNWRD_OC_LT = 0;
;  362          0489  3             RETURN
;  363          0490  2             END;
;  364          0491  2
;  365          0492  2         [PER_FLAG] :
;  366          0493  3             BEGIN
;  367          0494  3             !Capitalize flag
;  368          0495  3             SCA_WRD_LC_PNCT = .SCA_FILL;  !Double space only if .FILL.
;  369          0496  3             KCNS ();                      !Discard the flag.
;  370          0497  2             END;
;  371          0498  2
;  372          0499  2         [BRK_FLAG] :
;  373          0500  3             BEGIN
;  374          0501  3             !Break flag
;  375          0502  3
;  376          0503  3             KCNS ();                      !Skip over flag.
;  377          0504  3
;  378          0505  4             IF (.SCA_WRD_CPEND EQL RINTES)         !
;  379          0506  4                 OR (.KHAR EQL %C' ')               !Space?
;  380        0507  4                 OR (.KHAR EQL %O'011')                  !TAB??
;  381        0508  3             THEN
;  382        0509  3                 !'|' Before/after a space/tab or starting a line
;  383        0510  3                 ERMS  (RNFFNA, .FS_START (IRA), CH$DIFF (.FS_NEXT (IRA), .FS_START (IRA)))
;  384        0511  3             ELSE
;  385        0512  3
;  386        0513  3                 IF .KHAR EQL RINTES
;  387        0514  3                 THEN
;  388        0515  4                     BEGIN
;  389        0516  4                     ERM (RNFFEL, CH$PTR (UPLIT ('BREAK')), 5);
;  390        0517  4                     ERMS  (RNFSTR, .FS_START (IRA), CH$DIFF (.FS_NEXT (IRA), .FS_START (IRA)));
;  391        0518  4                     END
;  392        0519  3                 ELSE
;  393        0520  3                     !'|' after some text is ok.
;  394        0521  4                     BEGIN
;  395        0522  4                     ENDWRD (FALSE,FALSE, FALSE);          !End word with no spacing or justification.
;  396        0523  4                     SCA_FC_CASE = FALSE;                  !Do not cap next character.
;  397        0524  3                     END;
;  398        0525  2             END;
;  399        0526  2
;  400        0527  2         [CAP_FLAG] :
;  401        0528  3             BEGIN
;  402        0529  3             !Capitalize flag
;  403        0530  3
;  404        0531  3             IF .SCA_WORD_SET
;  405        0532  3             THEN
;  406        0533  3             !A Capitalize flag is already in effect.
;  407        0534  3             !Turn it off.  I.e. "<" is a flip-flop.
;  408        0535  4                 BEGIN                                   !Restore global case rules
;  409        0536  4                 SCA_WORD_SET = FALSE;
;  410        0537  4                 SCA_FC_UT = .SCA_FCBE_UT;
;  411        0538  4                 SCA_OC_UT = .SCA_OCBE_UT;
;  412        0539  4                 SCA_FC_LT = .SCA_FCBE_LT;
;  413        0540  4                 SCA_OC_LT = .SCA_OCBE_LT;
;  414        0541  4                 SCA_WRD_FC_UT = .SCA_FCBE_UT;
;  415        0542  4                 SCA_WRD_OC_UT = .SCA_OCBE_UT;
;  416        0543  4                 SCA_WRD_FC_LT = .SCA_FCBE_LT;
;  417        0544  4                 SCA_WRD_OC_LT = .SCA_OCBE_LT;
;  418        0545  4
;  419        0546  4                 SCA_MNFC_UT = .SCA_MNFCBE_UT;
;  420        0547  4                 SCA_MNOC_UT = .SCA_MNOCBE_UT;
;  421        0548  4                 SCA_MNFC_LT = .SCA_MNFCBE_LT;
;  422        0549  4                 SCA_MNOC_LT = .SCA_MNOCBE_LT;
;  423        0550  4                 SCA_MNWRD_FC_UT = .SCA_MNFCBE_UT;
;  424        0551  4                 SCA_MNWRD_OC_UT = .SCA_MNOCBE_UT;
;  425        0552  4                 SCA_MNWRD_FC_LT = .SCA_MNFCBE_LT;
;  426        0553  4                 SCA_MNWRD_OC_LT = .SCA_MNOCBE_LT
;  427        0554  4                 END
;  428        0555  3             ELSE                                        !
;  429        0556  4                 BEGIN
;  430        0557  4                 SCA_WORD_SET = TRUE;
;  431        0558  4                 SCA_FCBE_UT = .SCA_FC_UT;
;  432        0559  4                 SCA_FCBE_LT = .SCA_FC_LT;
;  433        0560  4                 SCA_OCBE_UT = .SCA_OC_UT;
;  434        0561  4                 SCA_OCBE_LT = .SCA_OC_LT;
;  435        0562  4
;  436        0563  4                 SCA_MNFCBE_UT = .SCA_MNFC_UT;
;  437      0564  4                           SCA_MNFCBE_LT = .SCA_MNFC_LT;
;  438      0565  4                           SCA_MNOCBE_UT = .SCA_MNOC_UT;
;  439      0566  4                           SCA_MNOCBE_LT = .SCA_MNOC_LT;
;  440      0567  4
;  441      0568  4                           ! Establish new case rules for this word.
;  442      0569  4                           SCA_WRD_FC_LT = %C'A' - %C'a';
;  443      0570  4                           SCA_WRD_OC_LT = %C'A' - %C'a';
;  444      0571  4                           SCA_WRD_FC_UT = 0;
;  445      0572  4                           SCA_WRD_OC_UT = 0;
;  446      0573  4                           SCA_FC_LT = %C'A' - %C'a';
;  447      0574  4                           SCA_OC_LT = %C'A' - %C'a';
;  448      0575  4                           SCA_FC_UT = 0;
;  449      0576  4                           SCA_OC_UT = 0;
;  450      0577  4
;  451      0578  4                           SCA_MNWRD_FC_LT = %DECIMAL'192' - %DECIMAL'224';
;  452      0579  4                           SCA_MNWRD_OC_LT = %DECIMAL'192' - %DECIMAL'224';
;  453      0580  4                           SCA_MNWRD_FC_UT = 0;
;  454      0581  4                           SCA_MNWRD_OC_UT = 0;
;  455      0582  4                           SCA_MNFC_LT = %DECIMAL'192' - %DECIMAL'224';
;  456      0583  4                           SCA_MNFC_UT = 0;
;  457      0584  4                           SCA_MNOC_LT = %DECIMAL'192' - %DECIMAL'224';
;  458      0585  4                           SCA_MNOC_UT = 0
;  459      0586  3                           END;
;  460      0587  3
;  461      0588  3                   KCNS ();
;  462      0589  3                   RETURN;
;  463      0590  2                   END;
;  464      0591  2
;  465      0592  2               [UND_FLAG] :
;  466      0593  3                   BEGIN
;  467      0594  3                   !Underline flag alone.
;  468      0595  3                   KCNS ();
;  469      0596  3                   IF .KHAR NEQ %O'011'                    !Next character a TAB?
;  470      0597  3                   THEN
;  471      0598  4                       BEGIN
;  472      0599  4                       SCA_WRD_AC_UND = .SCA_DO_UND;   ! Underline a single character
;  473      0600  3                       END;
;  474      0601  3
;  475      0602  3                   RETURN;
;  476      0603  2                   END;
;  477      0604  2
;  478      0605  2               [BLD_FLAG] :
;  479      0606  3                   BEGIN
;  480      0607  3                   !Bold flag alone.
;  481      0608  3                   KCNS ();
;  482      0609  3                   IF .KHAR NEQ %O'011'                    !Next character a TAB?
;  483      0610  3                   THEN
;  484      0611  4                       BEGIN
;  485      0612  4                       SCA_WRD_AC_BLD = .SCA_DO_BLD;   ! Bold a single character
;  486      0613  3                       END;
;  487      0614  3
;  488      0615  3                   RETURN;
;  489      0616  2                   END;
;  490      0617  2
;  491      0618  2               [SPA_FLAG] :
;  492      0619  3                   BEGIN
;  493      0620  3                   !Space flag
;  494     0621  3
;  495     0622  3              IF .SCA_X_FLAG
;  496     0623  3              THEN
;  497     0624  3              !A Space flag completes a sequence started by the Index flag.
;  498     0625  3                  XR ();
;  499     0626  3
;  500     0627  3              IF .SCA_WORD_SET
;  501     0628  3              THEN
;  502     0629  3              !A Space flag terminates a sequence started by
;  503     0630  3              !the Capitalize flag.
;  504     0631  4                  BEGIN                               !Restore global case rules
;  505     0632  4                  SCA_WORD_SET = FALSE;
;  506     0633  4                  SCA_FC_UT = .SCA_FCBE_UT;
;  507     0634  4                  SCA_OC_UT = .SCA_OCBE_UT;
;  508     0635  4                  SCA_FC_LT = .SCA_FCBE_LT;
;  509     0636  4                  SCA_OC_LT = .SCA_OCBE_LT;
;  510     0637  4                  SCA_WRD_FC_UT = .SCA_FCBE_UT;
;  511     0638  4                  SCA_WRD_OC_UT = .SCA_OCBE_UT;
;  512     0639  4                  SCA_WRD_FC_LT = .SCA_FCBE_LT;
;  513     0640  4                  SCA_WRD_OC_LT = .SCA_OCBE_LT;
;  514     0641  4
;  515     0642  4                  SCA_MNFC_UT = .SCA_MNFCBE_UT;
;  516     0643  4                  SCA_MNOC_UT = .SCA_MNOCBE_UT;
;  517     0644  4                  SCA_MNFC_LT = .SCA_MNFCBE_LT;
;  518     0645  4                  SCA_MNOC_LT = .SCA_MNOCBE_LT;
;  519     0646  4                  SCA_MNWRD_FC_UT = .SCA_MNFCBE_UT;
;  520     0647  4                  SCA_MNWRD_OC_UT = .SCA_MNOCBE_UT;
;  521     0648  4                  SCA_MNWRD_FC_LT = .SCA_MNFCBE_LT;
;  522     0649  4                  SCA_MNWRD_OC_LT = .SCA_MNOCBE_LT
;  523     0650  4
;  524     0651  3                  END;
;  525     0652  3
;  526     0653  3              SCA_WRD_C_UND = FALSE;                  !Do not underline this type of space unless forced.
;  527     0654  3              ENDCHR (%C' ');                         !Output a space
;  528     0655  3              SCA_FRC_CASE = FALSE;
;  529     0656  3              SCA_FC_CASE = TRUE;                     !Handle case as if a new word.
;  530     0657  3              KCNS ();
;  531     0658  3              RETURN;
;  532     0659  2              END;
;  533     0660  2
;  534     0661  2          [IND_FLAG] :
;  535     0662  3              BEGIN
;  536     0663  3              !Index flag
;  537     0664  3              IF .SCA_XROUTINE
;  538     0665  3              THEN
;  539     0666  3                  SUBXR ()
;  540     0667  3              ELSE
;  541     0668  3                  XR ();
;  542     0669  3              RETURN;
;  543     0670  2              END;
;  544     0671  2
;  545     0672  2          [HYP_FLAG] :
;  546     0673  3              BEGIN
;  547     0674  3              !Hyphenation flag
;  548     0675  3              KCNS ();                                !Skip flag.
;  549     0676  4              IF (.SCA_WRD_CPEND EQL RINTES)          !
;  550     0677  4                  OR (.KHAR EQL %C' ')                !Space?
```

```
:   551      0678  4                 OR (.KHAR EQL %O'011')                  !TAB??
:   552      0679  3             THEN
:   553      0680  3                 !Hyphenate before/after a space/tab or starting a line
:   554      0681  3                 ERMS  (RNFFNA, .FS_START (IRA), CH$DIFF (.FS_NEXT (IRA), .FS_START (IRA)))
:   555      0682  3             ELSE
:   556      0683  3
:   557      0684  3                 IF .KHAR EQL RINTES
:   558      0685  3                 THEN
:   559      0686  4                     BEGIN
:   560      0687  4                     ERM (RNFFEL, CH$PTR (UPLIT ('HYPHENATE')), 9);
:   561      0688  4                     ERMS  (RNFSTR, .FS_START (IRA), CH$DIFF (.FS_NEXT (IRA), .FS_START (IRA)));
:   562      0689  4                     END
:   563      0690  3                 ELSE
:   564      0691  3                     !Hyphenation between some text is ok.
:   565      0692  3                     IF .SCA_DO_HYP
:   566      0693  3                     THEN
:   567      0694  3                         !Hyphenation is only done if user said
:   568      0695  3                         !to do it (.HYPHENATION).  Otherwise it's ignored.
:   569      0696  4                         BEGIN
:   570      0697  4                         ENDWRD (FALSE,FALSE, TRUE);     !End word with hyphenation.
:   571      0698  4                         SCA_FC_CASE = FALSE;            !Do not cap next character.
:   572      0699  3                         END;
:   573      0700  3             RETURN;
:   574      0701  2             END;
:   575      0702  2
:   576      0703  2         [OVR_FLAG] :
:   577      0704  3             BEGIN
:   578      0705  3             !Overstrike flag
:   579      0706  3             KCNS ();
:   580      0707  3
:   581      0708  3             IF .KHAR EQL RINTES
:   582      0709  3             THEN
:   583      0710  4                 BEGIN
:   584      0711  4                 ERM (RNFFEL, CH$PTR (UPLIT ('OVERSTRIKE')), 10);
:   585      0712  4                 ERMS  (RNFSTR, .FS_START (IRA), CH$DIFF (.FS_NEXT (IRA), .FS_START (IRA)));
:   586      0713  4                 RETURN;         ! Ignore Overstrike flag at the end of a line.
:   587      0714  3                 END;
:   588      0715  3
:   589      0716  3             IF .SCA_WRD_CPEND EQL RINTES
:   590      0717  3             THEN
:   591      0718  4                 BEGIN
:   592      0719  4                 !OVERSTRIKE flag standing alone
:   593      0720  4                 ERMS  (RNFFNA, .FS_START (IRA), CH$DIFF (.FS_NEXT (IRA), .FS_START (IRA)));
:   594      0721  4                 END
:   595      0722  3             ELSE
:   596      0723  3                 IF .SCA_DO_OVR
:   597      0724  3                 THEN
:   598      0725  3                     !User did not say .NO OVERSTRIKING, so really overstrike.
:   599      0726  4                     BEGIN
:   600      0727  4                     IF (.FS_MAXSIZE (MRA) - .FS_LENGTH (MRA)) LSS 3
:   601      0728  4                     THEN        !No room for overstriking code
:   602      0729  4                         ERMS (RNFLTC, .FS_NEXT (IRA), .FS_LENGTH (IRA))
:   603      0730  4                     ELSE
:   604      0731  4                         !Generate overstriking code.
:   605      0732  5                         BEGIN
:   606      0733  5                         FS_WCHAR (MRA, RINTES);
:   607      0734  5                         FS_WCHAR (MRA, %C'O');
```

```
;   608          0735  5                             FS_WCHAR (MRA, .KHAR);
;   609          0736  5                             SCA_WRD_INT_L = .SCA_WRD_INT_L + 3;
;   610          0737  5                             SCA_WRD_OVR = TRUE
;   611          0738  4                             END;
;   612          0739  3                     END;
;   613          0740  3
;   614          0741  3             KCNS ();                                ! Get next character.
;   615          0742  3             RETURN;
;   616          0743  2             END;
;   617          0744  2
;   618        U 0745  2 %IF DSRPLUS %THEN
;   619        U 0746  2         [NPX_FLAG] :
;   620        U 0747  2             BEGIN
;   621        U 0748  2             !
;   622        U 0749  2             !If not an indexing command, then treat the Nopermute flag
;   623        U 0750  2             !as an ordinary character.
;   624        U 0751  2             IF NOT .TSF_INDEX
;   625        U 0752  2             THEN
;   626        U 0753  2                 BEGIN
;   627        U 0754  2                 ENDCHR (.KHAR);             !Output character as-is.
;   628        U 0755  2                 KCNS ();
;   629        U 0756  2                 RETURN;
;   630        U 0757  2                 END;
;   631        U 0758  2
;   632        U 0759  2             KCNS ();                        !Position past the flag.
;   633        U 0760  2
;   634        U 0761  2             IF .TSF_XYPLUS                  !Do no more unless this was an .XP/.YP command.
;   635        U 0762  2             THEN
;   636        U 0763  2                 BEGIN
;   637        U 0764  2                 IF (.SCA_WRD_CPEND NEQ RINTES)
;   638        U 0765  2                     OR (.KHAR EQL %C' ')                    !Space?
;   639        U 0766  2                     OR (.KHAR EQL %O'011')                  !Tab?
;   640        U 0767  2                 THEN
;   641        U 0768  2                     BEGIN
;   642        U 0769  2                     !Nopermute before a space/tab or in the middle of a word
;   643        U 0770  2                     ERMS (RNFFNA, .FS_START(IRA), CH$DIFF (.FS_NEXT(IRA), .FS_START(IRA)));
;   644        U 0771  2                     RETURN;
;   645        U 0772  2                     END;
;   646        U 0773  2
;   647        U 0774  2                 IF .KHAR EQL RINTES
;   648        U 0775  2                 THEN
;   649        U 0776  2                     BEGIN
;   650        U 0777  2                     ERM (RNFFEL, CH$PTR (UPLIT ('NOPERMUTE')), 9);
;   651        U 0778  2                     ERMS  (RNFSTR, .FS_START (IRA), CH$DIFF (.FS_NEXT (IRA), .FS_START (IRA)));
;   652        U 0779  2                     RETURN;
;   653        U 0780  2                     END;
;   654        U 0781  2
;   655        U 0782  2                 IF (.FS_MAXSIZE (MRA) - .FS_LENGTH (MRA)) LSS 3
;   656        U 0783  2                 THEN
;   657        U 0784  2                     BEGIN                   !No room for nopermute code.
;   658        U 0785  2                     ERMS (RNFLTC, .FS_NEXT (IRA), .FS_LENGTH (IRA));
;   659        U 0786  2                     RETURN;
;   660        U 0787  2                     END;
;   661        U 0788  2
;   662        U 0789  2                 !Now generate the no-permute escape sequence.
;   663        U 0790  2                 FS_WCHAR (MRA, RINTES);
;   664        U 0791  2                 FS_WCHAR (MRA, %C'P');
;  665      U 0792  2                FS_WCHAR (MRA, %C' ');
;  666      U 0793  2                SCA_WRD_INT_L = .SCA_WRD_INT_L + 3;
;  667      U 0794  2                END;
;  668      U 0795  2            END;
;  669        0796  2 %FI
;  670        0797  2          [INRANGE] :
;  671        0798  3              BEGIN
;  672        0799  3              KCNS ();
;  673        0800  3              RETURN;
;  674        0801  2              END;
;  675        0802  2
;  676        0803  2          [OUTRANGE] :
;  677        0804  3              BEGIN
;  678        0805  3              KCNS ();
;  679        0806  3              RETURN;
;  680        0807  2              END;
;  681        0808  2          TES;
;  682        0809  2
;  683        0810  2      RETURN;
;  684        0811  1      END;                                                  !End of DOFLG


                                                                      .TITLE  DOFLG Process flags
                                                                      .IDENT  \V04-000\

                                                                      .PSECT  $PLIT$,NOWRT,NOEXE,2

                   00  00  54  50  45  43  43  41  00000 P.AAA:  .ASCII  \ACCEPT\<0><0>                                   ;
                   00  00  00  4B  41  45  52  42  00008 P.AAB:  .ASCII  \BREAK\<0><0><0>                                 ;
   00  00  00  45  54  41  4E  45  48  50  59  48  00010 P.AAC:  .ASCII  \HYPHENATE\<0><0><0>                             ;
   00  00  45  4B  49  52  54  53  52  45  56  4F  0001C P.AAD:  .ASCII  \OVERSTRIKE\<0><0>                               ;

                                                                      .PSECT  $OWN$,NOEXE,2

                                                   00000 SCA_DO_HOLD:
                                                                      .BLKB   4
                                                   00004 SCA_WRD_CHOLD:
                                                                      .BLKB   4

                                                                      .EXTRN  RINTES, FLGT, IRA
                                                                      .EXTRN  KHAR, MRA, SCA, TSF
                                                                      .EXTRN  XMRA, XTSF, RNFFEL
                                                                      .EXTRN  RNFFNA, RNFLOC, RNFLTC
                                                                      .EXTRN  RNFSTR, ENDCHR, ENDWRD
                                                                      .EXTRN  ERM, ERMS, FNDFLG
                                                                      .EXTRN  OUTLIN, OUTXPH, SUBST
                                                                      .EXTRN  RSKIPS, SETCAS

                                                                      .PSECT  $CODE$,NOWRT,2

                                              0FFC 00000              .ENTRY  DOFLG, Save R2,R3,R4,R5,R6,R7,R8,R9,R10,R11 ; 0279
                           5B 00000000V   EF   9E 00002              MOVAB   XR, R11                                      ;
                           5A 00000000G   EF   9E 00009              MOVAB   ENDCHR, R10                                  ;
                           59 00000000G   EF   9E 00010              MOVAB   SETCAS, R9                                   ;
                           58 00000000'   EF   9E 00017              MOVAB   P.AAA, R8                                    ;
                           57 00000000G   EF   9E 0001E              MOVAB   FLGT+100, R7                                 ;
                           56        00G  8F   9A 00025              MOVZBL  #RINTES, R6                                  ;
```

```
                         55 00000000G  EF   9E 00029          MOVAB    KHAR, R5                        ;
                         54 00000000G  EF   9E 00030          MOVAB    IRA+12, R4                      ;
                         53 00000000G  EF   9E 00037          MOVAB    SCA+168, R3                     ;
                                            65   DD 0003E          PUSHL    KHAR                            ; 0309
                      00000000G  EF         01   FB 00040          CALLS    #1, FNDFLG                      ;
              11              00            50   CF 00047          CASEL    WHICH_FLAG, #0, #17             ; 0311
 002E         03E9         03E9       0026        0004B 1$:    .WORD    2$-1$,-                         ;
 0229         0184         00DC       005A        00053                 67$-1$,-                        ;
 02E7         02D7         0271       024F        0005B                 67$-1$,-                        ;
 0141         03E9         03E9       033D        00063                 3$-1$,-                         ;
                           03E9       014A        0006B                 7$-1$,-                         ;
                                                                        17$-1$,-                        ;
                                                                        34$-1$,-                        ;
                                                                        37$-1$,-                        ;
                                                                        40$-1$,-                        ;
                                                                        45$-1$,-                        ;
                                                                        48$-1$,-                        ;
                                                                        50$-1$,-                        ;
                                                                        58$-1$,-                        ;
                                                                        67$-1$,-                        ;
                                                                        67$-1$,-                        ;
                                                                        26$-1$,-                        ;
                                                                        28$-1$,-                        ;
                                                                        67$-1$                          ;
                                            79   11 0006F          BRB      13$                             ; 0805
                      00000000G  EF         00   FB 00071 2$:      CALLS    #0, SUBST                       ; 0320
                                                 04 00078          RET                                      ; 0315
                                            64   D5 00079 3$:      TSTL     IRA+12                          ; 0328
                                            08   14 0007B          BGTR     4$                              ;
                                65          56   9A 0007D          MOVZBL   R6, KHAR                        ;
                                64          01   CE 00080          MNEGL    #1, IRA+12                      ;
                                            09   11 00083          BRB      5$                              ;
                                65      F8  B4   9A 00085 4$:      MOVZBL   @IRA+4, KHAR                    ;
                                        F8  A4   D6 00089          INCL     IRA+4                           ;
                                            64   D7 0008C          DECL     IRA+12                          ;
                                56          65   D1 0008E 5$:      CMPL     KHAR, R6                        ; 0330
                                            07   12 00091          BNEQ     6$                              ;
                                            06   DD 00093          PUSHL    #6                              ; 0333
                                            58   DD 00095          PUSHL    R8                              ;
                                          030D   31 00097          BRW      61$                             ;
                             30  A3         01   D0 0009A 6$:      MOVL     #1, SCA+216                     ; 0339
                                            65   DD 0009E          PUSHL    KHAR                            ; 0340
                                 6A         01   FB 000A0          CALLS    #1, ENDCHR                      ;
                                            5B   11 000A3          BRB      15$                             ; 0341
                                            64   D5 000A5 7$:      TSTL     IRA+12                          ; 0350
                                            08   14 000A7          BGTR     8$                              ;
                                65          56   9A 000A9          MOVZBL   R6, KHAR                        ;
                                64          01   CE 000AC          MNEGL    #1, IRA+12                      ;
                                            09   11 000AF          BRB      9$                              ;
                                65      F8  B4   9A 000B1 8$:      MOVZBL   @IRA+4, KHAR                    ;
                                        F8  A4   D6 000B5          INCL     IRA+4                           ;
                                            64   D7 000B8          DECL     IRA+12                          ;
                             F4  A7         65   D1 000BA 9$:      CMPL     KHAR, FLGT+88                   ; 0352
                                            04   12 000BE          BNEQ     10$                             ;
                                            7E   D4 000C0          CLRL     -(SP)                           ; 0355
                                            0C   11 000C2          BRB      11$                             ;
                             FC  A7         65   D1 000C4 10$:     CMPL     KHAR, FLGT+96                   ; 0361
```

```
                                                  08  12 000C8           BNEQ    12$                                 ;
                                04          B4    A7  E9 000CA           BLBC    FLGT+24, 12$                        ; 0362
                                                  01  DD 000CE           PUSHL   #1                                  ; 0366
                                                  73  11 000D0 11$:      BRB     20$                                 ;
                                      67          65  D1 000D2 12$:      CMPL    KHAR, FLGT+100                      ; 0372
                                                  15  12 000D5           BNEQ    14$                                 ;
                                      11          B8    A7  E9 000D7           BLBC    FLGT+28, 14$                        ; 0373
                                F0    A3          02  88 000DB           BISB2   #2, SCA+152                         ; 0377
          50          63              01          01  EF 000DF           EXTZV   #1, #1, SCA+168, R0                 ; 0378
  1C      A3          01              01          50  F0 000E4           INSV    R0, #1, #1, SCA+196                 ;
                                                  73  11 000EA 13$:      BRB     22$                                 ; 0379
                                04    A7          65  D1 000EC 14$:      CMPL    KHAR, FLGT+104                      ; 0383
                                                  10  12 000F0           BNEQ    16$                                 ;
                                      0C          BC    A7  E9 000F2           BLBC    FLGT+32, 16$                        ; 0384
                                F0    A3          01  88 000F6           BISB2   #1, SCA+152                         ; 0388
  1C      A3          01              00          63  F0 000FA           INSV    SCA+168, #0, #1, SCA+196            ; 0389
                                                  71  11 00100 15$:      BRB     24$                                 ; 0390
                                30    A3          01  D0 00102 16$:      MOVL    #1, SCA+216                         ; 0412
                              FF6C    C3          20  CE 00106           MNEGL   #32, SCA+20                         ; 0413
                              FF74    C3          20  CE 0010B           MNEGL   #32, SCA+28                         ; 0414
                                          FF68    C3  D4 00110           CLRL    SCA+16                              ; 0415
                                          FF70    C3  D4 00114           CLRL    SCA+24                              ; 0416
                                9C    A3          20  CE 00118           MNEGL   #32, SCA+68                         ; 0418
                                A4    A3          20  CE 0011C           MNEGL   #32, SCA+76                         ; 0419
                                            98    A3  D4 00120           CLRL    SCA+64                              ; 0420
                                            A0    A3  D4 00123           CLRL    SCA+72                              ; 0421
                                                      04 00126           RET                                         ; 0348
                                                  64  D5 00127 17$:      TSTL    IRA+12                              ; 0428
                                                  08  14 00129           BGTR    18$                                 ;
                                      65          56  9A 0012B           MOVZBL  R6, KHAR                            ;
                                      64          01  CE 0012E           MNEGL   #1, IRA+12                          ;
                                                  09  11 00131           BRB     19$                                 ;
                                      65    F8    B4  9A 00133 18$:      MOVZBL  @IRA+4, KHAR                        ;
                                            F8    A4  D6 00137           INCL    IRA+4                               ;
                                                  64  D7 0013A           DECL    IRA+12                              ;
                                F8    A7          65  D1 0013C 19$:      CMPL    KHAR, FLGT+92                       ; 0430
                                                  0C  12 00140           BNEQ    21$                                 ;
                                      7E          01  CE 00142           MNEGL   #1, -(SP)                           ; 0433
                                      69          01  FB 00145 20$:      CALLS   #1, SETCAS                          ;
                                F8    A3          01  D0 00148           MOVL    #1, SCA+160                         ; 0434
                                                  44  11 0014C           BRB     27$                                 ; 0435
                                      67          65  D1 0014E 21$:      CMPL    KHAR, FLGT+100                      ; 0439
                                                  0E  12 00151           BNEQ    23$                                 ;
                                      0A    B8    A7  E9 00153           BLBC    FLGT+28, 23$                        ; 0440
                                F0    A3          02  8A 00157           BICB2   #2, SCA+152                         ; 0444
                                1C    A3          02  8A 0015B           BICB2   #2, SCA+196                         ; 0445
                                                  31  11 0015F 22$:      BRB     27$                                 ; 0446
                                04    A7          65  D1 00161 23$:      CMPL    KHAR, FLGT+104                      ; 0450
                                                  0E  12 00165           BNEQ    25$                                 ;
                                      0A    BC    A7  E9 00167           BLBC    FLGT+32, 25$                        ; 0451
                                F0    A3          01  8A 0016B           BICB2   #1, SCA+152                         ; 0455
                                1C    A3          01  8A 0016F           BICB2   #1, SCA+196                         ; 0456
                                                  1D  11 00173 24$:      BRB     27$                                 ; 0457
                                30    A3          01  D0 00175 25$:      MOVL    #1, SCA+216                         ; 0479
                              FF70    C3          20  7D 00179           MOVQ    #32, SCA+24                         ; 0481
                              FF68    C3          20  7D 0017E           MOVQ    #32, SCA+16                         ; 0480
                                A0    A3          20  7D 00183           MOVQ    #32, SCA+72                         ; 0486
```

```
    98  A3            20  7D 00187        MOVQ    #32, SCA+64             ; 0485
                          04 0018B        RET                             ; 0426
  00A0  C3      C0    B3  D0 0018C 26$:   MOVL    @SCA+104, SCA+328       ; 0495
                    029F  31 00192 27$:   BRW     67$                     ; 0496
                      64  D5 00195 28$:   TSTL    IRA+12                  ; 0503
                      08  14 00197        BGTR    29$                     ;
        65            56  9A 00199        MOVZBL  R6, KHAR                ;
        64            01  CE 0019C        MNEGL   #1, IRA+12              ;
                      09  11 0019F        BRB     30$                     ;
        65      F8    B4  9A 001A1 29$:   MOVZBL  @IRA+4, KHAR            ;
                F8    A4  D6 001A5        INCL    IRA+4                   ;
                      64  D7 001A8        DECL    IRA+12                  ;
        56      70    A3  D1 001AA 30$:   CMPL    SCA+280, R6             ; 0505
                      08  13 001AE        BEQL    31$                     ;
        20            65  D1 001B0        CMPL    KHAR, #32               ; 0506
                      03  13 001B3        BEQL    31$                     ;
        09            65  D1 001B5        CMPL    KHAR, #9                ; 0507
                      03  12 001B8 31$:   BNEQ    32$                     ;
                    019A  31 001BA        BRW     53$                     ;
        56            65  D1 001BD 32$:   CMPL    KHAR, R6                ; 0513
                      08  12 001C0        BNEQ    33$                     ;
                      05  DD 001C2        PUSHL   #5                      ; 0516
                08    A8  9F 001C4        PUSHAB  P.AAB                   ;
                    01DD  31 001C7        BRW     61$                     ;
                      7E  D4 001CA 33$:   CLRL    -(SP)                   ; 0522
                    01AC  31 001CC        BRW     57$                     ;
        4A      B8    A3  E9 001CF 34$:   BLBC    SCA+96, 35$             ; 0531
                B8    A3  D4 001D3        CLRL    SCA+96                  ; 0536
  FF58  C3    FF78    C3  7D 001D6        MOVQ    SCA+32, SCA             ; 0537
  FF60  C3      80    A3  7D 001DD        MOVQ    SCA+40, SCA+8           ; 0539
  FF68  C3    FF78    C3  D0 001E3        MOVL    SCA+32, SCA+16          ; 0541
  FF70  C3    FF7C    C3  D0 001EA        MOVL    SCA+36, SCA+24          ; 0542
  FF6C  C3      80    A3  D0 001F1        MOVL    SCA+40, SCA+20          ; 0543
  FF74  C3      84    A3  D0 001F7        MOVL    SCA+44, SCA+28          ; 0544
    88  A3      A8    A3  7D 001FD        MOVQ    SCA+80, SCA+48          ; 0546
    90  A3      B0    A3  7D 00202        MOVQ    SCA+88, SCA+56          ; 0548
    98  A3      A8    A3  D0 00207        MOVL    SCA+80, SCA+64          ; 0550
    A0  A3      AC    A3  D0 0020C        MOVL    SCA+84, SCA+72          ; 0551
    9C  A3      B0    A3  D0 00211        MOVL    SCA+88, SCA+68          ; 0552
    A4  A3      B4    A3  D0 00216        MOVL    SCA+92, SCA+76          ; 0553
                      54  11 0021B        BRB     36$                     ;
    B8  A3            01  D0 0021D 35$:   MOVL    #1, SCA+96              ; 0557
    80  A3    FF60    C3  7D 00221        MOVQ    SCA+8, SCA+40           ; 0559
  FF78  C3    FF58    C3  7D 00227        MOVQ    SCA, SCA+32             ; 0558
    B0  A3      90    A3  7D 0022E        MOVQ    SCA+56, SCA+88          ; 0564
    A8  A3      88    A3  7D 00233        MOVQ    SCA+48, SCA+80          ; 0563
  FF6C  C3            20  CE 00238        MNEGL   #32, SCA+20             ; 0569
  FF74  C3            20  CE 0023D        MNEGL   #32, SCA+28             ; 0570
              FF68    C3  D4 00242        CLRL    SCA+16                  ; 0571
              FF70    C3  D4 00246        CLRL    SCA+24                  ; 0572
  FF60  C3            20  CE 0024A        MNEGL   #32, SCA+8              ; 0573
  FF64  C3            20  CE 0024F        MNEGL   #32, SCA+12             ; 0574
              FF58    C3  7C 00254        CLRQ    SCA                     ; 0575
    9C  A3            20  CE 00258        MNEGL   #32, SCA+68             ; 0578
    A4  A3            20  CE 0025C        MNEGL   #32, SCA+76             ; 0579
                98    A3  D4 00260        CLRL    SCA+64                  ; 0580
                A0    A3  D4 00263        CLRL    SCA+72                  ; 0581
```

```
                                                             I  6
DOFLG            Process flags                          16-Sep-1984 00:13:49   VAX-11 Bliss-32 V4.0-742                Page  18
V04-000          Module Level Declarations              14-Sep-1984 13:06:00   DISK$VMSMASTER:[RUNOFF.SRC]DOFLG.BLI;1    (4)

                                    90    A3             20 CE 00266           MNEGL   #32, SCA+56                             ; 0582
                                    94    A3             20 CE 0026A           MNEGL   #32, SCA+60                             ; 0584
                                                   88    A3 7C 0026E           CLRQ    SCA+48                                  ; 0583
                                                       01C0 31 00271 36$:      BRW     67$                                     ; 0588
                                                         64 D5 00274 37$:      TSTL    IRA+12                                  ; 0595
                                                         08 14 00276           BGTR    38$                                     ;
                                          65             56 9A 00278           MOVZBL  R6, KHAR                                ;
                                          64             01 CE 0027B           MNEGL   #1, IRA+12                              ;
                                                         09 11 0027E           BRB     39$                                     ;
                                          65       F8    B4 9A 00280 38$:      MOVZBL  @IRA+4, KHAR                            ;
                                                   F8    A4 D6 00284           INCL    IRA+4                                   ;
                                                         64 D7 00287           DECL    IRA+12                                  ;
                                          09             65 D1 00289 39$:      CMPL    KHAR, #9                                ; 0596
                                                         24 13 0028C           BEQL    43$                                     ;
            50             63             01             01 EF 0028E           EXTZV   #1, #1, SCA+168, R0                     ; 0599
       20   A3             01             01             50 F0 00293           INSV    R0, #1, #1, SCA+200                     ;
                                                         04 00299              RET                                             ; 0593
                                                         64 D5 0029A 40$:      TSTL    IRA+12                                  ; 0608
                                                         08 14 0029C           BGTR    41$                                     ;
                                          65             56 9A 0029E           MOVZBL  R6, KHAR                                ;
                                          64             01 CE 002A1           MNEGL   #1, IRA+12                              ;
                                                         09 11 002A4           BRB     42$                                     ;
                                          65       F8    B4 9A 002A6 41$:      MOVZBL  @IRA+4, KHAR                            ;
                                                   F8    A4 D6 002AA           INCL    IRA+4                                   ;
                                                         64 D7 002AD           DECL    IRA+12                                  ;
                                          09             65 D1 002AF 42$:      CMPL    KHAR, #9                                ; 0609
                                                         01 12 002B2 43$:      BNEQ    44$                                     ;
                                                         04 002B4              RET                                             ;
       20   A3             01             00             63 F0 002B5 44$:      INSV    SCA+168, #0, #1, SCA+200                ; 0612
                                                         04 002BB              RET                                             ; 0606
                                          03       F4    A3 E9 002BC 45$:      BLBC    SCA+156, 46$                            ; 0622
                                          6B             00 FB 002C0           CALLS   #0, XR                                  ; 0625
                                          48       B8    A3 E9 002C3 46$:      BLBC    SCA+96, 47$                             ; 0627
                                                   B8    A3 D4 002C7           CLRL    SCA+96                                  ; 0632
                                  FF58    C3     FF78    C3 7D 002CA           MOVQ    SCA+32, SCA                             ; 0633
                                  FF60    C3       80    A3 7D 002D1           MOVQ    SCA+40, SCA+8                           ; 0635
                                  FF68    C3     FF78    C3 D0 002D7           MOVL    SCA+32, SCA+16                          ; 0637
                                  FF70    C3     FF7C    C3 D0 002DE           MOVL    SCA+36, SCA+24                          ; 0638
                                  FF6C    C3       80    A3 D0 002E5           MOVL    SCA+40, SCA+20                          ; 0639
                                  FF74    C3       84    A3 D0 002EB           MOVL    SCA+44, SCA+28                          ; 0640
                                    88    A3       A8    A3 7D 002F1           MOVQ    SCA+80, SCA+48                          ; 0642
                                    90    A3       B0    A3 7D 002F6           MOVQ    SCA+88, SCA+56                          ; 0644
                                    98    A3       A8    A3 D0 002FB           MOVL    SCA+80, SCA+64                          ; 0646
                                    A0    A3       AC    A3 D0 00300           MOVL    SCA+84, SCA+72                          ; 0647
                                    9C    A3       B0    A3 D0 00305           MOVL    SCA+88, SCA+68                          ; 0648
                                    A4    A3       B4    A3 D0 0030A           MOVL    SCA+92, SCA+76                          ; 0649
                                    1C    A3             02 8A 0030F 47$:      BICB2   #2, SCA+196                             ; 0653
                                                         20 DD 00313           PUSHL   #32                                     ; 0654
                                          6A             01 FB 00315           CALLS   #1, ENDCHR                              ;
                                                   F8    A3 D4 00318           CLRL    SCA+160                                 ; 0655
                                    28    A3             01 D0 0031B           MOVL    #1, SCA+208                             ; 0656
                                                       0112 31 0031F           BRW     67$                                     ; 0657
                                          08       10    A3 E9 00322 48$:      BLBC    SCA+184, 49$                            ; 0664
                             00000000V    EF             00 FB 00326           CALLS   #0, SUBXR                               ; 0666
                                                         04 0032D              RET                                             ;
                                          6B             00 FB 0032E 49$:      CALLS   #0, XR                                  ; 0668
                                                         04 00331              RET                                             ; 0662
```

```
                                64  D5 00332 50$:    TSTL     IRA+12                ; 0675
                                08  14 00334         BGTR     51$                   ;
                 65             56  9A 00336         MOVZBL   R6, KHAR              ;
                 64             01  CE 00339         MNEGL    #1, IRA+12            ;
                                09  11 0033C         BRB      52$                   ;
                 65         F8  B4  9A 0033E 51$:    MOVZBL   @IRA+4, KHAR          ;
                            F8  A4  D6 00342         INCL     IRA+4                 ;
                                64  D7 00345         DECL     IRA+12                ;
                 56         70  A3  D1 00347 52$:    CMPL     SCA+280, R6           ; 0676
                                0A  13 0034B         BEQL     53$                   ;
                 20             65  D1 0034D         CMPL     KHAR, #32             ; 0677
                                05  13 00350         BEQL     53$                   ;
                 09             65  D1 00352         CMPL     KHAR, #9              ; 0678
                                11  12 00355         BNEQ     54$                   ;
7E           F8  A4         F4  A4  C3 00357 53$:    SUBL3    IRA, IRA+4, -(SP)     ; 0681
                            F4  A4  DD 0035D         PUSHL    IRA                   ;
                     00000000G  8F  DD 00360         PUSHL    #RNFFNA               ;
                                5B  11 00366         BRB      62$                   ;
                 56             65  D1 00368 54$:    CMPL     KHAR, R6              ; 0684
                                07  12 0036B         BNEQ     55$                   ;
                                09  DD 0036D         PUSHL    #9                    ; 0687
                            10  A8  9F 0036F         PUSHAB   P.AAC                 ;
                                33  11 00372         BRB      61$                   ;
01               63             04  E0 00374 55$:    BBS      #4, SCA+168, 56$      ; 0692
                                    04 00378         RET                            ;
                                01  DD 00379 56$:    PUSHL    #1                    ; 0697
                                7E  7C 0037B 57$:    CLRQ     -(SP)                 ;
      00000000G  EF             03  FB 0037D         CALLS    #3, ENDWRD            ;
                            28  A3  D4 00384         CLRL     SCA+208               ; 0698
                                    04 00387         RET                            ; 0673
                                64  D5 00388 58$:    TSTL     IRA+12                ; 0706
                                08  14 0038A         BGTR     59$                   ;
                 65             56  9A 0038C         MOVZBL   R6, KHAR              ;
                 64             01  CE 0038F         MNEGL    #1, IRA+12            ;
                                09  11 00392         BRB      60$                   ;
                 65         F8  B4  9A 00394 59$:    MOVZBL   @IRA+4, KHAR          ;
                            F8  A4  D6 00398         INCL     IRA+4                 ;
                                64  D7 0039B         DECL     IRA+12                ;
                 56             65  D1 0039D 60$:    CMPL     KHAR, R6              ; 0708
                                29  12 003A0         BNEQ     63$                   ;
                                0A  DD 003A2         PUSHL    #10                   ; 0711
                            10  A8  9F 003A4         PUSHAB   P.AAD                 ;
                     00000000G  8F  DD 003A7 61$:    PUSHL    #RNFFEL               ;
      00000000G  EF             03  FB 003AD         CALLS    #3, ERM               ;
7E           F8  A4         F4  A4  C3 003B4         SUBL3    IRA, IRA+4, -(SP)     ; 0712
                            F4  A4  DD 003BA         PUSHL    IRA                   ;
                     00000000G  8F  DD 003BD         PUSHL    #RNFSTR               ;
      00000000G  EF             03  FB 003C3 62$:    CALLS    #3, ERMS              ;
                                    04 003CA         RET                            ; 0710
                 56         70  A3  D1 003CB 63$:    CMPL     SCA+280, R6           ; 0716
                                11  12 003CF         BNEQ     64$                   ;
7E           F8  A4         F4  A4  C3 003D1         SUBL3    IRA, IRA+4, -(SP)     ; 0720
                            F4  A4  DD 003D7         PUSHL    IRA                   ;
                     00000000G  8F  DD 003DA         PUSHL    #RNFFNA               ;
                                21  11 003E0         BRB      65$                   ;
4E               63             02  E1 003E2 64$:    BBC      #2, SCA+168, 67$      ; 0723
                 52  00000000G  EF  D0 003E6         MOVL     MRA, R2               ; 0727
```

```
                      50        0C    A2           03  C1 003ED        ADDL3   #3, 12(R2), R0              ;
                                      50     08    A2  D1 003F2        CMPL    8(R2), R0                   ;
                                                   14  18 003F6        BGEQ    66$                         ;
                                                   64  DD 003F8        PUSHL   IRA+12                      ; 0729
                                             F8    A4  DD 003FA        PUSHL   IRA+4                       ;
                                      00000000G    8F  DD 003FD        PUSHL   #RNFLTC                     ;
                         00000000G    EF           03  FB 00403  65$:  CALLS   #3, ERMS                    ;
                                                   28  11 0040A        BRB     67$                         ;
                                      50     04    A2  9E 0040C  66$:  MOVAB   4(R2), R0                   ; 0733
                                00    B0           56  90 00410        MOVB    R6, @0(R0)                  ;
                                                   60  D6 00414        INCL    (R0)                        ;
                                             0C    A2  D6 00416        INCL    12(R2)                      ;
                                00    B0     4F    8F  90 00419        MOVB    #79, @0(R0)                 ; 0734
                                                   60  D6 0041E        INCL    (R0)                        ;
                                             0C    A2  D6 00420        INCL    12(R2)                      ;
                                00    B0           65  90 00423        MOVB    KHAR, @0(R0)                ; 0735
                                                   60  D6 00427        INCL    (R0)                        ;
                                             0C    A2  D6 00429        INCL    12(R2)                      ;
                                54    A3           03  C0 0042C        ADDL2   #3, SCA+252                 ; 0736
                                18    A3           04  88 00430        BISB2   #4, SCA+192                 ; 0737
                                                   64  D5 00434  67$:  TSTL    IRA+12                      ; 0799
                                                   07  14 00436        BGTR    68$                         ;
                                      65           56  9A 00438        MOVZBL  R6, KHAR                    ;
                                      64           01  CE 0043B        MNEGL   #1, IRA+12                  ;
                                                       04 0043E        RET                                 ;
                                      65     F8    B4  9A 0043F  68$:  MOVZBL  @IRA+4, KHAR                ;
                                             F8    A4  D6 00443        INCL    IRA+4                       ;
                                                   64  D7 00446        DECL    IRA+12                      ;
                                                       04 00448        RET                                 ; 0811

; Routine Size:  1097 bytes,    Routine Base:  $CODE$ + 0000
;   686     0812  1 GLOBAL ROUTINE SUBXR : NOVALUE =
;   687     0813  1
;   688     0814  1 !++
;   689     0815  1 ! FUNCTIONAL DESCRIPTION:
;   690     0816  1 !
;   691     0817  1 !       This routine is called when RUNOFF is processing a .SUBINDEX or
;   692     0818  1 !       .ENTRY command, and a Subindex flag is encountered. The character
;   693     0819  1 !       sequence that the previous Subindex flag started is terminated, and the
;   694     0820  1 !       next one is set up.
;   695     0821  1 !
;   696     0822  1 ! FORMAL PARAMETERS:  None
;   697     0823  1 !
;   698     0824  1 ! IMPLICIT INPUTS: None
;   699     0825  1 !
;   700     0826  1 ! IMPLICIT OUTPUTS: None
;   701     0827  1 !
;   702     0828  1 ! ROUTINE VALUE:
;   703     0829  1 ! COMPLETION CODES: None
;   704     0830  1 !
;   705     0831  1 ! SIDE EFFECTS: None
;   706     0832  1 !
;   707     0833  1 !--
;   708     0834  1
;   709     0835  2     BEGIN
;   710     0836  2     !Always position past the Subindex flag
;   711     0837  2     KCNS ();
;   712     0838  2     !Skip spaces/tabs preceeding the next item.
;   713     0839  2     RSKIPS (IRA);
;   714     0840  2
;   715     0841  2     !Make sure user didn't put the Subindex flag at the end of
;   716     0842  2     !the line.
;   717     0843  2     IF .KHAR EQL RINTES                         !End of line?
;   718     0844  2     THEN
;   719     0845  3         BEGIN
;   720     0846  3         ERM (RNFFEL, CH$PTR (UPLIT ('SUBINDEX')), 8);
;   721     0847  3         ERMS  (RNFSTR, .FS_START (IRA), CH$DIFF (.FS_NEXT (IRA), .FS_START (IRA)) - 1);
;   722     0848  3         RETURN
;   723     0849  2         END;
;   724     0850  2
;   725     0851  2     !Make sure user didn't put the Subindex flag at the start of the line.
;   726     0852  2     !The check for the start of the line is done by seeing if there
;   727     0853  2     !is anything in the index output buffer yet, as would be the case
;   728     0854  2     !if the two pointers are indeed equal.  On top of that, it's
;   729     0855  2     !important to take account of single-character entires.  Remember
;   730     0856  2     !that ENDCHR won't put a character into the MRA until the next
;   731     0857  2     !character forces it out.  So checking for it being the first character
;   732     0858  2     !detects the one-character-entry case.
;   733     0859  2     IF .SCA_FC                                  !First character?
;   734     0860  3         AND (.FS_START (MRA) EQL .FS_NEXT (MRA))!Start of line?
;   735     0861  2     THEN
;   736     0862  3         BEGIN
;   737     0863  3         ERMS (RNFFNA,
;   738     0864  3             .FS_START (IRA),
;   739     0865  3             CH$DIFF (.FS_NEXT (IRA), .FS_START (IRA)) - 1);
;   740     0866  3         RETURN
;   741     0867  2         END;
;   742     0868  2
```

```
2	!Detect ">>" and complain.
2	!The check is done by looking back at what's been generated to see if
2	!all there is is a justification mark, to separate words.  Similarily,
2	!the case of a one-character entry is taken into account.
3	BEGIN
3	LOCAL
3	    TEMP_PTR;
3	TEMP_PTR = CH$PLUS(.FS_NEXT (MRA), -3);		!Point to start of a sequence.
4	IF (CH$RCHAR_A (TEMP_PTR) EQL RINTES)		!
4	    AND (CH$RCHAR (.TEMP_PTR) EQL %C'J')		!
3	    AND .SCA_FC
3	THEN
4	    BEGIN
4	    ERMS (RNFFNA,
4		.FS_START (IRA),
4		CH$DIFF (.FS_NEXT (IRA), .FS_START (IRA)) - 1);
4	    RETURN
4	    END
2	END;
2
2	IF .SCA_WRD_CPEND NEQ RINTES
2	THEN
2	    ENDWRD (FALSE, FALSE, FALSE)
2	ELSE
2	!Dump trailing spaces
3	    BEGIN
3	    LOCAL
3		TEMP_PTR;
3
3	    FS_NEXT (MRA) = CH$PLUS (.FS_NEXT (MRA), -.SCA_WRD_LST_SP);
3	    FS_LENGTH (MRA) = .FS_LENGTH (MRA) - .SCA_WRD_LST_SP;
3	    SCA_WRD_LST_SP = 0;
3
3	    !Check for wierd things like .X ^^(spaces only), or .X a>^^(spaces)>etc
3	    TEMP_PTR = CH$PLUS(.FS_NEXT (MRA), -3); !Point to start of a sequence.
5	    IF ((CH$RCHAR_A (TEMP_PTR) EQL RINTES)		!
4		AND (CH$RCHAR (.TEMP_PTR) EQL %C'J'))		!Catches .X whatever>^^(spaces)>whatever
4		OR (.FS_START(MRA) EQL .FS_NEXT(MRA))		!Catches .X^^(spaces)>whatever
3	    THEN
4		BEGIN
4		ERMS (RNFFNA,
4		    .FS_START (IRA),
4		    CH$DIFF (.FS_NEXT (IRA), .FS_START (IRA)) - 1);
4		RETURN
4		END
2	    END;
2
2	!An end of word marker separates character sequences.
2	FS_WCHAR (MRA, RINTES);
2	FS_WCHAR (MRA, %C'J');
2	FS_WCHAR (MRA, %C' ');
2	TSF_INT_HL = .TSF_INT_HL + 3;
2	TSF_JUS_CNT = .TSF_JUS_CNT + 1;
2	SCA_WRD_PNTR = .FS_NEXT (MRA)
1	END;					!End of SUBXR
```

```
                                              .PSECT  $PLIT$,NOWRT,NOEXE,2

  58  45  44  4E  49  42  55  53 00028 P.AAE: .ASCII  \SUBINDEX\                           ;

                                              .PSECT  $CODE$,NOWRT,2

                            00FC 00000        .ENTRY  SUBXR, Save R2,R3,R4,R5,R6,R7         ; 0812
              57 00000000G EF 9E 00002        MOVAB   KHAR, R7                              ;
              56 00000000G EF 9E 00009        MOVAB   MRA, R6                               ;
              55       00G 8F 9A 00010        MOVZBL  #RINTES, R5                           ;
              54 00000000G EF 9E 00014        MOVAB   SCA+332, R4                           ;
              53 00000000G EF 9E 0001B        MOVAB   IRA, R3                               ;
                        0C A3 D5 00022        TSTL    IRA+12                                ; 0837
                           09 14 00025        BGTR    1$                                    ;
              67           55 9A 00027        MOVZBL  R5, KHAR                              ;
           0C A3           01 CE 0002A        MNEGL   #1, IRA+12                            ;
                           0A 11 0002E        BRB     2$                                    ;
              67        04 B3 9A 00030 1$:    MOVZBL  @IRA+4, KHAR                          ;
                        04 A3 D6 00034        INCL    IRA+4                                 ;
                        0C A3 D7 00037        DECL    IRA+12                                ;
                           53 DD 0003A 2$:    PUSHL   R3                                    ; 0839
    00000000G EF           01 FB 0003C        CALLS   #1, RSKIPS                            ;
              55           67 D1 00043        CMPL    KHAR, R5                              ; 0843
                           27 12 00046        BNEQ    3$                                    ;
                           08 DD 00048        PUSHL   #8                                    ; 0846
                 00000000' EF 9F 0004A        PUSHAB  P.AAE                                 ;
                 00000000G 8F DD 00050        PUSHL   #RNFFEL                               ;
    00000000G EF           03 FB 00056        CALLS   #3, ERM                               ;
50         04 A3           63 C3 0005D        SUBL3   IRA, IRA+4, R0                        ; 0847
                        FF A0 9F 00062        PUSHAB  -1(R0)                                ;
                           63 DD 00065        PUSHL   IRA                                   ;
                 00000000G 8F DD 00067        PUSHL   #RNFSTR                               ;
                           72 11 0006D        BRB     9$                                    ;
              09      FF48 C4 E9 0006F 3$:    BLBC    SCA+148, 4$                           ; 0859
              50           66 D0 00074        MOVL    MRA, R0                               ; 0860
           04 A0           60 D1 00077        CMPL    (R0), 4(R0)                           ;
                           54 13 0007B        BEQL    8$                                    ;
              50           66 D0 0007D 4$:    MOVL    MRA, R0                               ; 0876
50         04 A0           03 C3 00080        SUBL3   #3, 4(R0), TEMP_PTR                   ;
              51           80 9A 00085        MOVZBL  (TEMP_PTR)+, R1                       ; 0877
              55           51 D1 00088        CMPL    R1, R5                                ;
                           0B 12 0008B        BNEQ    5$                                    ;
           4A 8F           60 91 0008D        CMPB    (TEMP_PTR), #74                       ; 0878
                           05 12 00091        BNEQ    5$                                    ;
              39      FF48 C4 E8 00093        BLBS    SCA+148, 8$                           ; 0879
              55        CC A4 D1 00098 5$:    CMPL    SCA+280, R5                           ; 0889
                           0D 13 0009C        BEQL    6$                                    ;
                           7E 7C 0009E        CLRQ    -(SP)                                 ; 0891
                           7E D4 000A0        CLRL    -(SP)                                 ;
    00000000G EF           03 FB 000A2        CALLS   #3, ENDWRD                            ;
                           3E 11 000A9        BRB     10$                                   ;
              50           66 D0 000AB 6$:    MOVL    MRA, R0                               ; 0898
           04 A0           64 C2 000AE        SUBL2   SCA+332, 4(R0)                        ;
           0C A0           64 C2 000B2        SUBL2   SCA+332, 12(R0)                       ; 0899
```

```
                                            64  D4 000B6         CLRL    SCA+332                                  ; 0900
                  51      04    A0          03  C3 000B8         SUBL3   #3, 4(R0), TEMP_PTR                      ; 0903
                                52          81  9A 000BD         MOVZBL  (TEMP_PTR)+, R2                          ; 0904
                                55          52  D1 000C0         CMPL    R2, R5                                   ;
                                            06  12 000C3         BNEQ    7$                                       ;
                          4A    8F          61  91 000C5         CMPB    (TEMP_PTR), #74                          ; 0905
                                            06  13 000C9         BEQL    8$                                       ;
                          04    A0          60  D1 000CB 7$:     CMPL    (R0), 4(R0)                              ; 0906
                                            18  12 000CF         BNEQ    10$                                      ;
                  50      04    A3          63  C3 000D1 8$:     SUBL3   IRA, IRA+4, R0                           ; 0911
                                       FF   A0  9F 000D6         PUSHAB  -1(R0)                                   ;
                                            63  DD 000D9         PUSHL   IRA                                      ; 0910
                                00000000G   8F  DD 000DB         PUSHL   #RNFFNA                                  ; 0909
                    00000000G   EF          03  FB 000E1 9$:     CALLS   #3, ERMS                                 ;
                                                04 000E8         RET                                              ; 0908
                                50          66  D0 000E9 10$:    MOVL    MRA, R0                                  ; 0917
                                51     04   A0  9E 000EC         MOVAB   4(R0), R1                                ;
                          00    B1          55  90 000F0         MOVB    R5, @0(R1)                               ;
                                            61  D6 000F4         INCL    (R1)                                     ;
                                       0C   A0  D6 000F6         INCL    12(R0)                                   ;
                          00    B1     4A   8F  90 000F9         MOVB    #74, @0(R1)                              ; 0918
                                            61  D6 000FE         INCL    (R1)                                     ;
                                       0C   A0  D6 00100         INCL    12(R0)                                   ;
                          00    B1          20  90 00103         MOVB    #32, @0(R1)                              ; 0919
                                            61  D6 00107         INCL    (R1)                                     ;
                                       0C   A0  D6 00109         INCL    12(R0)                                   ;
                                50 00000000G EF D0 0010C         MOVL    TSF, R0                                  ;
                                60          03  C0 00113         ADDL2   #3, (R0)                                 ; 0920
                                       20   A0  D6 00116         INCL    32(R0)                                   ; 0921
                          AC    A4          61  D0 00119         MOVL    (R1), SCA+248                            ; 0922
                                                04 0011D         RET                                              ; 0923

; Routine Size:  286 bytes,    Routine Base:  $CODE$ + 0449
```

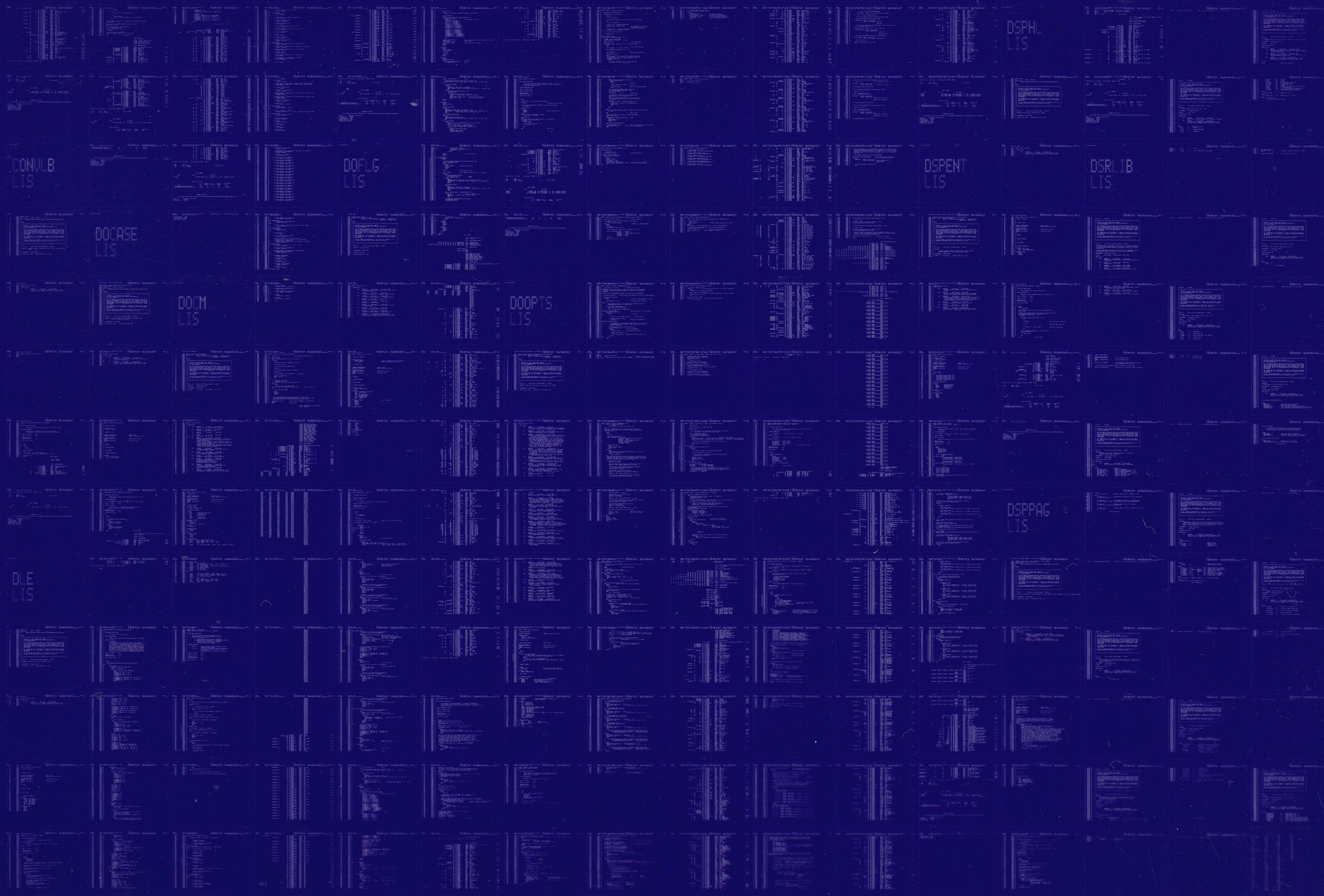
```
1 GLOBAL ROUTINE XR : NOVALUE =
1
1 !++
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       This routine is called when an Index flag is seen. It is also
1 !       called when a space is seen at the end of a word, and that word is
1 !       to be entered into the index.
1 !
1 ! FORMAL PARAMETERS: None
1 !
1 ! IMPLICIT INPUTS: None
1 !
1 ! IMPLICIT OUTPUTS: None
1 !
1 ! ROUTINE VALUE:
1 ! COMPLETION CODES: None
1 !
1 ! SIDE EFFECTS: None
1 !
1 !--
1
2     BEGIN
2
2     IF .KHAR EQL .FLGT [IND_FLAG, FLAG_CHARACTER]
2     THEN
2         KCNS ();                              !Position past Index flag
2
2     !If indexing is already on, terminate current phrase and enter
2     !it into the idex.
2
2     IF .SCA_X_FLAG
2     THEN
3         BEGIN
3         OUTXPH ();                            !Output word to index.
3         RETURN;
3         END
2     ELSE
2     !Turn on indexing if indexing is wanted.
2         SCA_X_FLAG = .SCA_DO_IND;
2
2     IF .SCA_X_FLAG
2     THEN
2     !The index flag has just been turned on.  Initialize the
2     !work areas for the Index flag.
3         BEGIN
3         FS_INIT (XMRA);
3
3         INCR I FROM 0 TO TSF_SIZE - 1 DO
3             XTSF [.I] = 0;
3
2         END;
2
1     END;                                      !End of XR
```

```
                                                   003C 00000       .ENTRY  XR, Save R2,R3,R4,R5                        ; 0924
                          55 00000000G   EF    9E 00002             MOVAB   KHAR, R5                                    ;
                          54 00000000G   EF    9E 00009             MOVAB   SCA+156, R4                                 ;
                          53 00000000G   EF    9E 00010             MOVAB   XMRA, R3                                    ;
                          52 00000000G   EF    9E 00017             MOVAB   IRA+12, R2                                  ;
               00000000G  EF             65    D1 0001E             CMPL    KHAR, FLGT+112                              ; 0948
                                         16    12 00025             BNEQ    2$                                          ;
                                         62    D5 00027             TSTL    IRA+12                                      ; 0950
                                         09    14 00029             BGTR    1$                                          ;
                          65        00G  8F    9A 0002B             MOVZBL  #RINTES, KHAR                               ;
                          62             01    CE 0002F             MNEGL   #1, IRA+12                                  ;
                                         09    11 00032             BRB     2$                                          ;
                          65        F8   B2    9A 00034  1$:        MOVZBL  @IRA+4, KHAR                                ;
                                    F8   A2    D6 00038             INCL    IRA+4                                       ;
                                         62    D7 0003B             DECL    IRA+12                                      ;
                          08             64    E9 0003D  2$:        BLBC    SCA+156, 3$                                 ; 0955
               00000000G  EF             00    FB 00040             CALLS   #0, OUTXPH                                  ; 0958
                                         04       00047             RET                                                 ; 0957
     64       0C    A4                   01    03 EF 00048  3$:     EXTZV   #3, #1, SCA+168, SCA+156                    ; 0963
                          18             64    E9 0004E             BLBC    SCA+156, 5$                                 ; 0965
                                    0C   A3    D4 00051             CLRL    XMRA+12                                     ; 0970
                          63        10   A3    9E 00054             MOVAB   XMRA+16, XMRA                               ;
                    04    A3             63    D0 00058             MOVL    XMRA, XMRA+4                                ;
                                         50    D4 0005C             CLRL    I                                           ; 0972
                          00000000GEF40        D4 0005E  4$:        CLRL    XTSF[I]                                     ; 0973
               F5         50             27    F3 00065             AOBLEQ  #39, I, 4$                                  ;
                                               04 00069  5$:        RET                                                 ; 0977

; Routine Size:  106 bytes,    Routine Base:  $CODE$ + 0567


;  853          0978  1
;  854          0979  1 END                                         !End of module
;  855          0980  0 ELUDOM




;                                PSECT SUMMARY
;
;     Name                     Bytes                    Attributes
;
;  $OWN$                           8  NOVEC,  WRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $PLIT$                         48  NOVEC,NOWRT,  RD ,NOEXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;  $CODE$                       1489  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)




;                          Library Statistics
;
;                                    -------- Symbols --------      Pages       Processing
;     File                           Total    Loaded   Percent      Mapped      Time
```

```
;
;  _$255$DUA28:[SYSLIB]XPORT.L32;1                 590          0           0         252            00:00.1
;  _$255$DUA28:[RUNOFF.SRC]DSRLIB.L32;1           1248        100           8          86            00:00.3




;                                         COMMAND QUALIFIERS

;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS$:DOFLG/OBJ=OBJ$:DOFLG MSRC$:DOFLG/UPDATE=(ENH$:DOFLG)

; Size:           1489 code + 56 data bytes
; Run Time:          00:40.3
; Elapsed Time:      01:29.0
; Lines/CPU Min:       1457
; Lexemes/CPU-Min: 24624
; Memory Used:  360 pages
; Compilation Complete
```

0339 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY



DSPHL
LIS

CONVLB
LIS

DOFLG
LIS

DSPENT
LIS

DSRLIB
LIS

DOCASE
LIS

DOCM
LIS

DOOPTS
LIS

DSPPAG
LIS

DLE
LIS